

Knight Foundation School of Computing and Information Sciences

Spring 2023 Senior Design Project



Knight Foundation
School of Computing
and Information Sciences

Analyzing Biological Networks

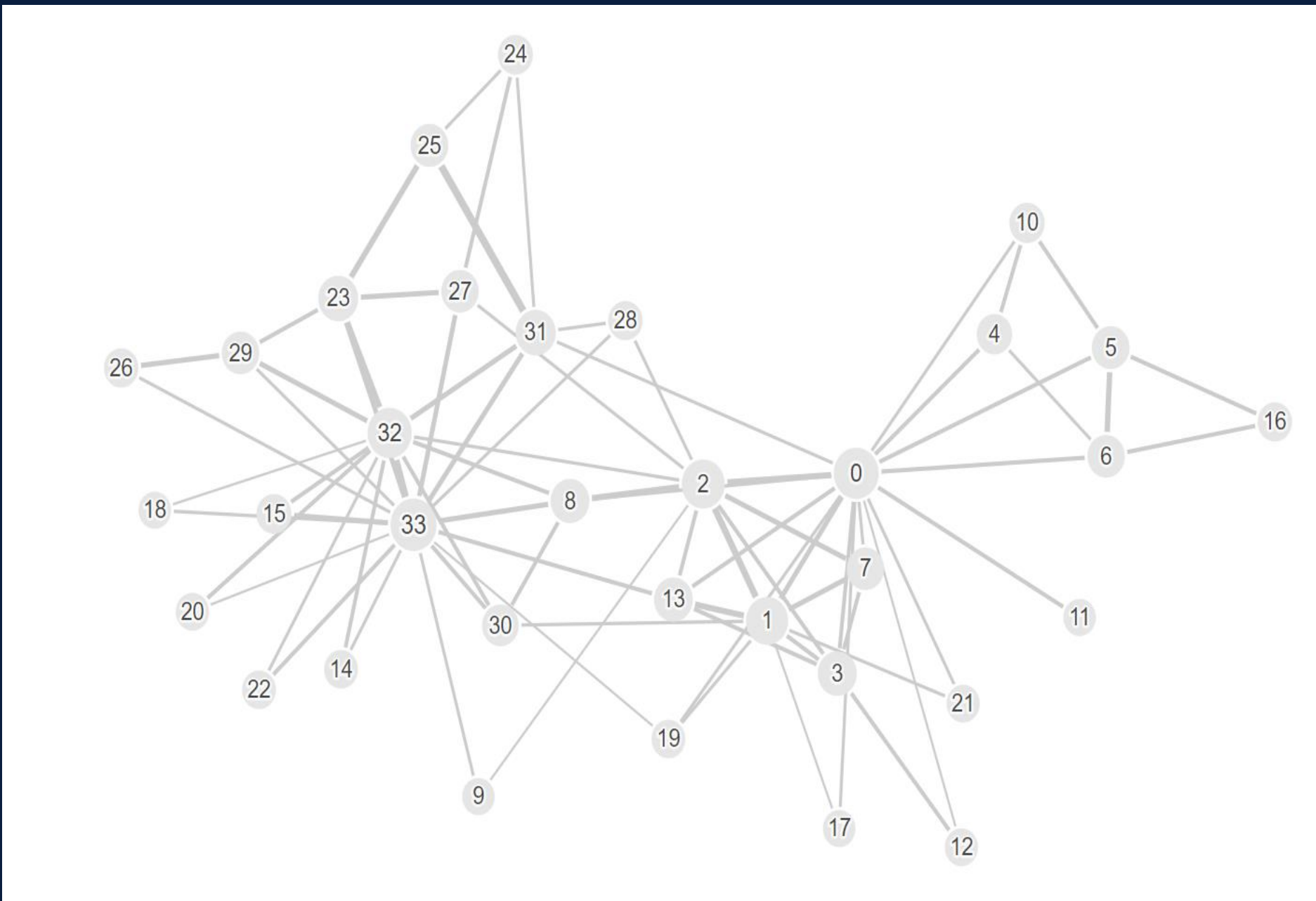
Students: Osmany Pujol, Vyacheslav Lazurenko, Anthony Abello, Sage Pages, Kyle Gareth Brindle

Mentor: Raihanul Bari Tanvir , Md Mezbahul Islam

Instructor/Faculty: Alex Afanasyev, Florida International University

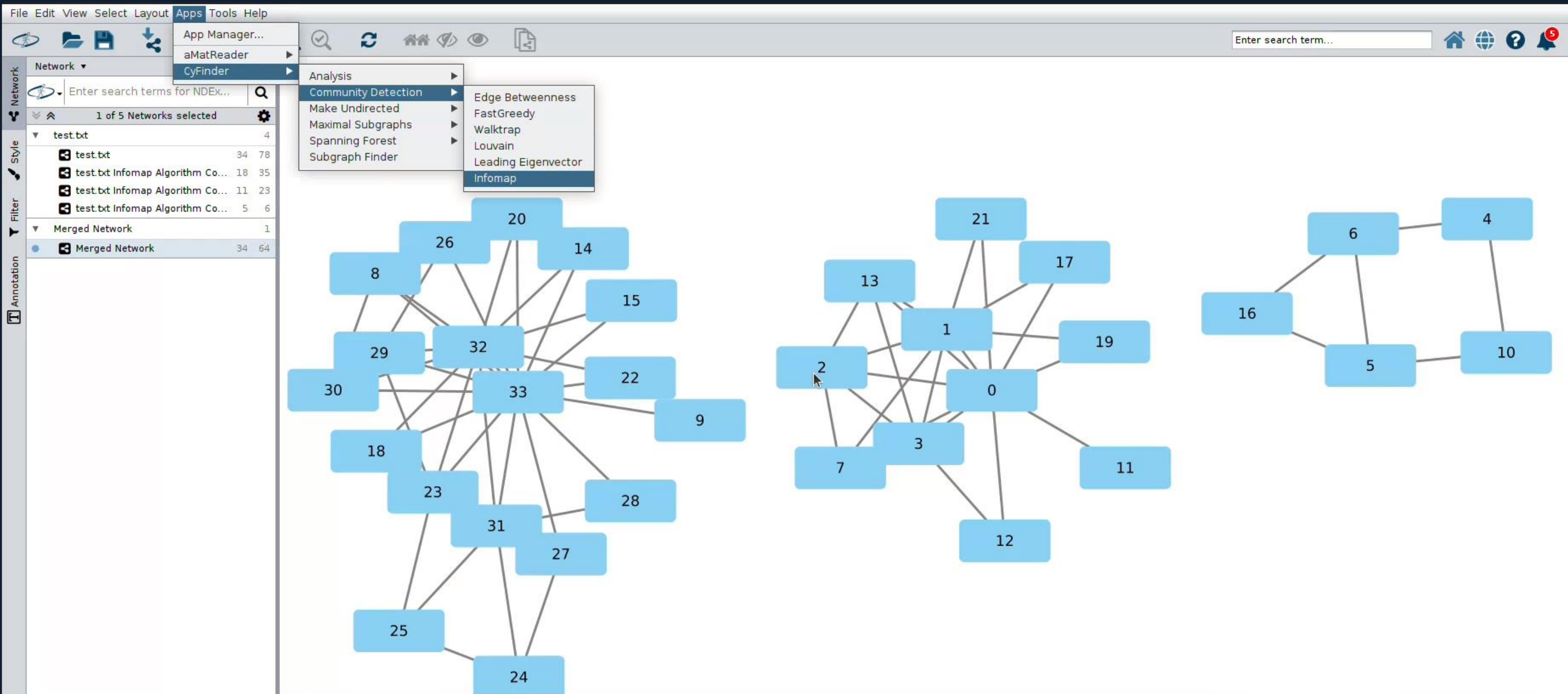
PROBLEM

Scientists and researchers are interested in **investigating biological networks** through the application of **community detection algorithms**. Although **Cytoscape** is an open-source tool commonly employed for this purpose, it may not encompass all available algorithms, particularly newer ones.

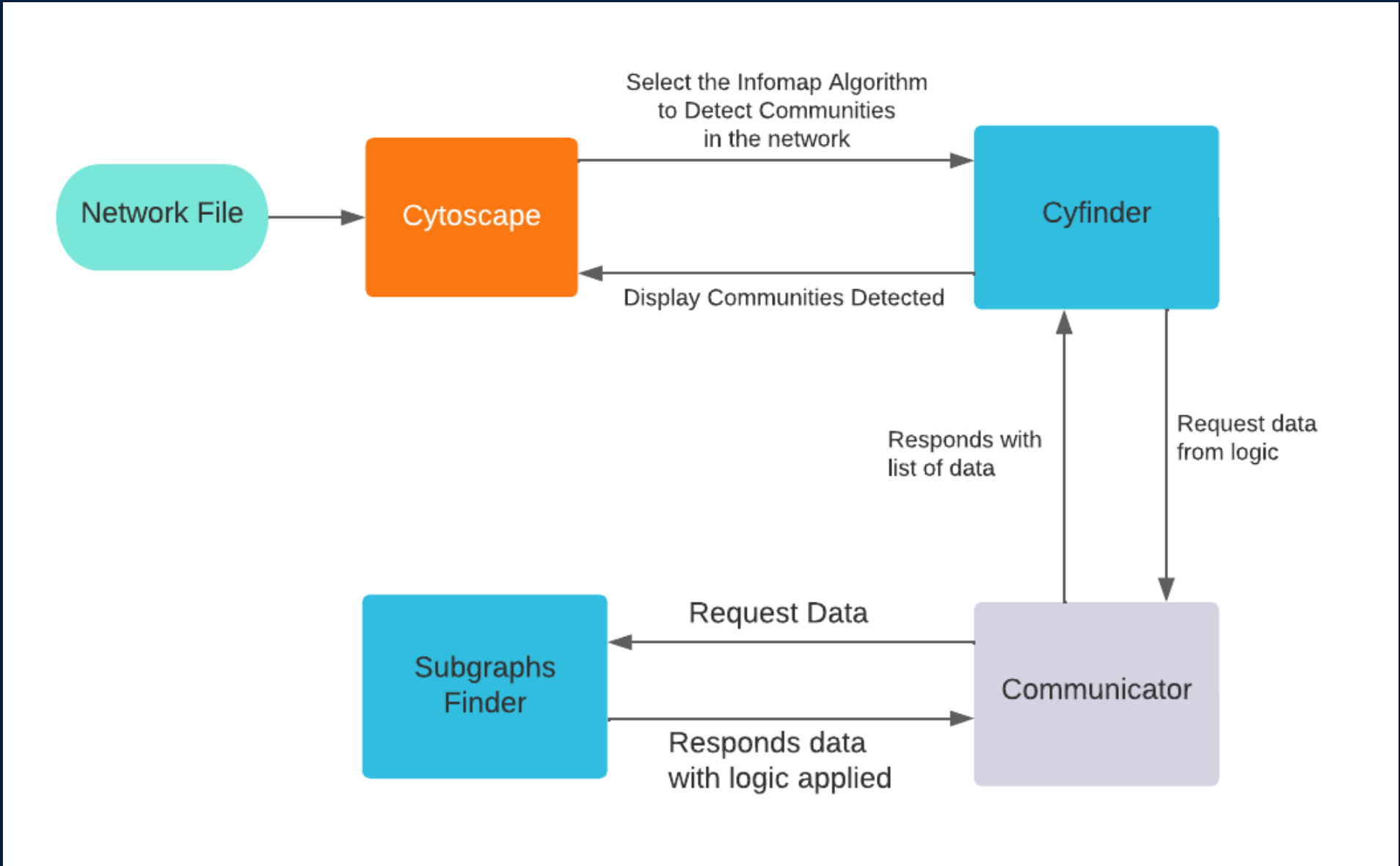


SOLUTION

The **Louvain** method and **Infomap** algorithm are among the most used **community detection** algorithms in network analysis. The Louvain method prioritizes modularity optimization, while Infomap utilizes **information theory**. In our recent project, we have concentrated on implementing and adding the Infomap algorithm to the **Cyfinder** plugin for Cytoscape.



SYSTEM DESIGN



IMPLEMENTATION

To detect communities Infomap uses the Map Equation developed by the researchers to calculate the average bit length needed to represent the network (MDL). By minimizing the total description of the network, we can obtain the most meaningful community structure.

$$L(M) = w_{\alpha} \log(w_{\alpha}) - 2 \sum_{i=1}^m w_{i\alpha} \log(w_{i\alpha}) - \sum_{\alpha=1}^n w_{\alpha} \log(w_{\alpha}) + \sum_{i=1}^m (w_{i\alpha} + w_i) \log(w_{i\alpha} + w_i)$$

Map Equation for an Undirected Weighted graph

Symbol meanings:
 α : current node in the graph
 i : the module to which node α is part of
 w_{α} : the relative weight of the node's links
 w_i : is the sum of the relative weights of every node in the module i
 $w_{i\alpha}$: relative weight of the links exiting the module
 $w_{\alpha i}$: total relative weight of links between modules

```
while (shouldContinue) {
    shouldContinue = false;
    for (Node node : nodeList) {
        int currentModule = getModule(node);
        int bestModule = currentModule;
        double bestDelta = 0.0;
        // try moving this node to every neighbor's module and identify the best option
        for (Node neighbor : node.getNeighbors()) {
            int neighborModule = getModule(neighbor);
            if (neighborModule == currentModule) {
                continue;
            }
            double delta = modularityGain(node, currentModule, neighborModule);
            if (delta > bestDelta) {
                bestDelta = delta;
                bestModule = neighborModule;
            }
        }
        // move node to the best module option, don't move if current module is the best
        if (bestModule != currentModule) {
            shouldContinue = true;
            moveNode(node, bestModule);
        }
    }
}
```

- Main steps of the InfomapAlgorithm:
1. Assign every **Node** to its own **Module**
 2. Pick a **random node** in the network, and one by one, assign its **neighboring Node** to its **Module**
 3. DO FOR EVERY NEIGHBOR:
 1. Calculate the average bit length of the network using the Map Equation
 4. Calculate the modularity gain by comparing the average bit length of both networks, the one before moving the Node to the Module and the one after.
 5. Assign the neighboring node that contributes the highest modularity gain.
 6. Continue doing this until the modularity does not improve.

Requirements



Java 11.0
Main language used to implement the entire project.



Maven
Used to package the entire Cyfinder application into a jar file installed into Cytoscape.



Apache Ant
Subgraphs Finder uses ant to build all the logic into a jar file that is used by Cyfinder.



Cytoscape 3.9.1
The software for which we built the community detection algorithm and visualizing networks.

SUMMARY

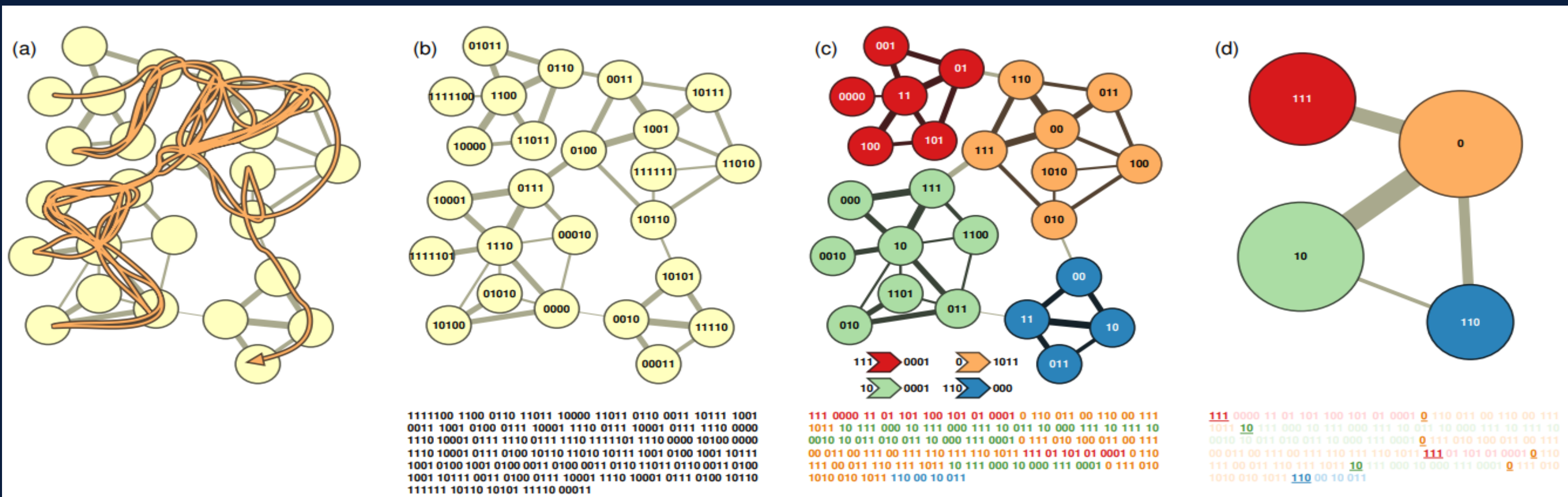
- We **conducted research** and **developed** the **Infomap** community detection algorithm for **undirected weighted graphs**, drawing inspiration from the scholarly work [Mapping Higher-Order Network Flows in Memory and Multilayer Networks with Infomap].
- Subsequently, we integrated this algorithm into the **GUI** and connected it to **Cytoscape**, as demonstrated in the proposed solution.
- Although the aforementioned paper outlines several further **optimizations** that could expedite the **algorithm's performance**, time constraints limited our ability to incorporate these modifications.
- Furthermore, the algorithm can also be **adapted** for use in undirected **unweighted and weighted graphs**, making it a versatile tool for community detection.

ACKNOWLEDGEMENT

The material presented in this poster is based upon the work supported by Ananda Mondal. I thank Ananda Mondal , Md Mezbahul Islam and Raihanul Bari Tanvir for their assistance and mentorship that we received throughout the senior design project.

REFERENCES

Edler, D., Bohlin, L., & Rosvall, M. (2017). Mapping Higher-Order Network Flows in Memory and Multilayer Networks with Infomap. Article <https://www.mapequation.org/infomap>



Engineering
& Computing